UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/755,525	01/05/2001	Daniel Melchione	NETAP005	5062

28875 7590 09/02/2004

Zilka-Kotab, PC
P.O. BOX 721120
SAN JOSE, CA 95172-1120

EXAMINER	
CHOUDHURY, AZIZUL Q	

ART UNIT	PAPER NUMBER
2143	

DATE MAILED: 09/02/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/755,525	**Applicant(s)** MELCHIONE ET AL.	
	Examiner Azizul Choudhury	**Art Unit** 2143	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 May 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-38* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-38* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *05 January 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date *1/13/03, 3/12/01*.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Detailed Action

This office action is in response to the amendment received May 28, 2004.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1, 6-9, 11-14, 18-25, and 29-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Barroux (US Pat No: US005923850A) in view of Ries et al (US Pat No: US006061724), hereafter referred to as Barroux and Ries, respectively.

1. With regards to claim 1, Barroux teaches through Ries, a system for management of a network of devices and resources available to the devices via a computer network, comprising: a network directory defining a network topology of nodes corresponding to the network of devices and defining policies corresponding to and to be enforced upon the resources available to the devices; a policy orchestrator

server in communication with the network directory, the policy orchestrator server being adapted to determine a hierarchical tree structure containing the nodes based upon location of each node in the network topology, determine a policy for each node in the hierarchical tree structure, and communicate said policy to the corresponding node; and an agent corresponding to each device in the network of devices, the agent being in communication with the policy orchestrator server and the resources corresponding to the device, the agent being adapted to receive data from the policy orchestrator server and to enforce the policies corresponding to the resources, wherein the policies corresponding to the resources of each device are selectively inherited along the hierarchical tree structure; wherein a management console is in communication with the network directory and the policy orchestrator server for providing a user interface, the management console being adapted to selectively display the hierarchical tree structure and the policies stored in the network directory; wherein the management console dynamically determines the policy of a selected node of the hierarchical tree structure; wherein the management console dynamically determines the policy of the selected node by reading the policies of nodes along a path of nodes from a root of the hierarchical tree structure to the selected node and wherein the management console overwrites previously written policies upon reading conflicting policies at each node along the path of nodes; wherein each agent includes a task execution component for causing performance of a scheduled task at a scheduled time, a policy enforcement component for enforcing policies applicable to resources of the corresponding device, a property collection component for collecting and storing properties of the corresponding resources of the device and for transmitting the properties to the policy orchestrator server, and an event collection component for collecting and storing event data and for transmitting the event data to the policy orchestrator server

(Barroux teaches a system for managing a network of devices and resources. A GUI is provided to view the network from a hierarchical level (topology) (column 4, lines 49-52, Barroux). Barroux's design also allows for policy defining and enforcing (column 3, line 64 – column 4, line 14, Barroux). This includes the search of the network and policy handling techniques claimed. In addition, Barroux's design has agents within each device to assist in monitoring and managing the network (column 3, lines 41-53, Barroux). Plus, Barroux teaches how the GUI is an interface for managing (management console) (column 3, lines 24-40, Barroux). Barroux also teaches how the processes to be performed, are computed (column 4, line 3, Barroux), hence they are determined dynamically. In addition, it is inherent that nodes will be check along the hierarchical path. Barroux's design checks nodes individually along the network path (column 14, lines 25 – column 18, line 21, Barroux). Finally, Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. In addition, Barroux's design allows for scheduled tasks (column 3, line 64 – column 4, line 13, Barroux) and device property retrieval and storage as claimed (column 14, line 25 – column 18, line 21, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

2. With regards to claim 6, Barroux teaches through Ries, the system for management of a network of devices and resources via a computer network, wherein each policy is selected from the group consisting of a configuration rule and a scheduled task

(Barroux teaches a system for managing a network of devices and resources. The administrative database contains a database of the policies that are enforceable and hence the policy can be selected as claimed (column 3, line 54 – column 4, line 10, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

3. With regards go claim 7, Barroux teaches through Ries, the system for management of a network of devices and resources via a computer network, wherein at least one policy is a scheduled task and wherein the agent causes performance of the task when the scheduled task is to be performed

(Barroux teaches a system for managing a network of devices and resources. The policies of Barroux's design are scheduled tasks (column 3, line 54 – column 4, line 10, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

4. With regards to claim 8, Barroux teaches through Ries, the system for management of a network of devices and resources via a computer network, wherein the policy orchestrator server includes a software repository adapted to be selectively, transmitted to the device via the agent corresponding to the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

5. With regards to claim 9, Barroux teaches through Ries, the system for management of a network of devices and resources via a computer network, wherein the policy orchestrator server includes an agent installation module adapted to be transmitted to the device for installation of the corresponding agent on the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

5. With regards to claim 11, Barroux teaches through Ries a method for management of a network of devices and resources available to the devices via a computer network, comprising: determining a hierarchical tree structure based upon locations of devices in a network topology, each device being a node in the hierarchical tree structure; determining policies for each node in the hierarchical tree structure to be enforced by an agent corresponding to each node, the agent being in communication with the device and the resources corresponding to the device; and communicating the policy to the corresponding agent; and wherein the policies

corresponding to the resources of each device are selectively inherited along the hierarchical tree structure of the network directory; wherein the policy of a selected node of the hierarchical tree structure is dynamically determined by the management console; wherein the dynamic determining includes reading policies of nodes along a path of nodes from a root of the hierarchical tree structure down to the selected node and overwriting previously written policies upon reading conflicting policies at each node along the path of nodes; wherein each agent includes a task execution component for causing performance of a scheduled task at a scheduled time, a policy enforcement component for enforcing policies applicable to resources of the corresponding device, a property collection component for collecting and storing properties of the corresponding resources of the device and for transmitting the properties to the policy orchestrator server, and an event collection component for collecting and storing event data and for transmitting the event data to the policy orchestrator server

(Barroux teaches a system for managing a network of devices and resources. A GUI is provided to view the network from a hierarchical level (topology) (column 4, lines 49-52, Barroux). Barroux's design also allows for policy defining and enforcing (column 3, line 64 – column 4, line 14, Barroux). This includes the search of the network and policy handling techniques claimed. In addition, Barroux's design has agents within each device to assist in monitoring and managing the network (column 3, lines 41-53, Barroux). Plus the processes to be performed, are computed (column 4, line 3, Barroux), hence they are determined dynamically. In addition, it is inherent that nodes will be check along the hierarchical path. Barroux's design checks nodes individually along the network path (column 14, lines 25 – column 18, line 21, Barroux). Finally, Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. In addition, Barroux's design allows for scheduled tasks (column 3, line 64 – column 4, line 13, Barroux) and device property retrieval and storage as claimed (column 14, line 25 – column 18, line 21, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

6. With regards to claim 12, Barroux teaches through Reis, the method for management of a network of devices and resources via a computer network, further comprising enforcing the polices by the agent upon the corresponding device and resources available to the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has the policies enforce by agents, as in all network monitoring or managing systems (column 3, line 24 – column 4, line 19, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

7. With regards to claim 13, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, further comprising: transmitting an agent installation package to the corresponding device; installing the agent on the device; and transmitting a message by the agent to the policy orchestrator server, the message containing properties of the device, the agent being adapted to enforce the policies corresponding to the resources contained in the network directory

(Barroux teaches a system for managing a network of devices and resources. The agents in network monitoring/managing systems have to be installed, as they are in Barroux's design and as claimed (column 11, lines 40-48, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

8. With regards to claim 14, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, further comprising selectively displaying the hierarchical tree structure and the policies stored in the network directory by a management console in communication with the network directory and the policy orchestrator server, the management console providing a user interface

(Barroux teaches a system for managing a network of devices and resources. A GUI is provided to view the network from a hierarchical level (topology) (column 4, lines 49-52, Barroux). In addition, Barroux teaches how the GUI is a an interface for managing (management console) (column 3, lines 24-40, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

9. With regards to claim 18, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, wherein each policy is selected from the group consisting of a configuration rule and a scheduled task

(Barroux teaches a system for managing a network of devices and resources. The administrative database contains a database of the policies that are enforceable and hence the policy can be selected as claimed (column 3, line 54 – column 4, line 10, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

10. With regards to claim 19, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, wherein when the policy is a scheduled task, further comprising causing performance of the task by the agent when the scheduled task is to be performed

(Barroux teaches a system for managing a network of devices and resources. The administrative database contains a database of the policies that are enforceable and hence the policy can be selected as claimed (column 3, line 54 – column 4, line 10, Barroux). The task inherently is performed as claimed, or else there is no purpose to having the task. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

11. With regards to claim 20, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, further comprising selectively transmitting data from a software repository of the policy orchestrator server to the device via the agent corresponding to the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has with software transferred to the network device through the agent (column 11, line 40 – column 14, line 23, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

12. With regards to claim 21, Barroux teaches the method for management of a network of devices and resources via a computer network, further comprising transmitting an agent installation module by the policy orchestrator server to the device for installation of the corresponding agent on the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

6. With regards to claim 22, Barroux teaches through Ries, a computer program product for management of a network of devices and resources available to the devices via a computer network, comprising: computer code that determines a hierarchical tree structure based upon locations of devices in a network topology, each device being a node in the hierarchical tree structure; computer code that determines policies for each node in the hierarchical tree structure to be enforced by an agent corresponding to each node, the agent being in communication with the device and the resources corresponding to the device; computer code that

communicates the policy to the corresponding agent; wherein the policies corresponding to the resources of each device are selectively inherited along the hierarchical tree structure of the network directory; computer code that dynamically determines the policy of a selected node of the hierarchical tree structure by the management console; wherein the computer code that dynamically determines includes computer code that reads policies of nodes along a path of nodes from a path of nodes from a root of the hierarchical tree structure down to the selected node and computer code that overwrites previously written policies upon reading conflicting policies at each node along the path of nodes; wherein each agent includes a task execution component for causing performance of a scheduled task at a scheduled time, a policy enforcement component for enforcing policies applicable to resources of the corresponding device, a property collection component for collecting and storing properties of the corresponding resources of the device and for transmitting the properties to the policy orchestrator server, and an event collection component for collecting and storing event data and for transmitting the event data to the policy orchestrator server; and a computer readable medium that stores said computer codes;

(Barroux teaches a system for managing a network of devices and resources. A GUI is provided to view the network from a hierarchical level (topology) (column 4, lines 49-52, Barroux). Barroux's design also allows for policy defining and enforcing (column 3, line 64 – column 4, line 14, Barroux). This includes the search of the network and policy handling techniques claimed. In addition, Barroux's design has agents within each device to assist in monitoring and managing the network (column 3, lines 41-53, Barroux). Plus in Barroux's design the processes to be performed, are computed (column 4, line 3, Barroux), hence they are determined dynamically. In addition, it is inherent that nodes will be check along the hierarchical path. Barroux's design checks nodes individually along the network path (column 14, lines 25 – column 18, line 21, Barroux). Finally, Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. In addition, Barroux's design allows for scheduled tasks (column 3, line 64 – column 4, line 13, Barroux) and device property retrieval and storage as claimed (column 14, line 25 – column 18, line 21, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

13. With regards to claim 23, Barroux teaches through Ries, the computer program product for management of a network of devices and resources via a computer network, further comprising computer code that enforces the polices upon the corresponding device and resources available to the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has the policies enforce by agents, as in all network monitoring or managing systems (column 3, line 24 – column 4, line 19, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

14. With regards to claim 24, Barroux teaches through Ries, the computer program product for management of a network of devices and resources via a computer network, further comprising: computer code that transmits an agent installation package to the corresponding device; computer code that installs the agent on the device; and computer code that transmits a message by the agent to the policy

orchestrator server, the message containing properties of the device, the agent being adapted to enforce the policies corresponding to the resources contained in the network directory

(Barroux teaches a system for managing a network of devices and resources. The agents in network monitoring/managing systems have to be installed, as they are in Barroux's design and as claimed (column 11, lines 40-48, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

15. With regard to claim 25, Barroux teaches through Ries, the computer program product for management of a network of devices and resources via a computer network, further comprising computer code that selectively displays the hierarchical tree structure and the policies stored in the network directory in communication with the network directory and the policy orchestrator server, the management console providing a user interface

(Barroux teaches a system for managing a network of devices and resources. A GUI is provided to view the network from a hierarchical level (topology) (column 4, lines 49-52, Barroux). In addition, Barroux teaches how the GUI is a an interface for managing (management console) (column 3, lines 24-40, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

16. With regards to claim 29, Barroux teaches through Ries, the computer program product for management of a network of devices and resources via a computer network, wherein each policy is selected from the group consisting of a configuration rule and a scheduled task

(Barroux teaches a system for managing a network of devices and resources. The administrative database contains a database of the policies that are enforceable and hence the policy can be selected as claimed (column 3, line 54 – column 4, line 10, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

17. With regards to claim 30, Barroux teaches through Ries, the computer program product for management of a network of devices and resources via a computer network, wherein when the policy is a scheduled task, further comprising computer code that causes performance of the task by the agent when the scheduled task is to be performed

(Barroux teaches a system for managing a network of devices and resources. The administrative database contains a database of the policies that are enforceable and hence the policy can be selected as claimed (column 3, line 54 – column 4, line 10, Barroux). The task inherently is performed as claimed, or else there is no purpose to having the task. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

18. With regards to claim 31, Barroux teaches through Ries, the computer program product for management of a network of devices and resources via a computer network, further comprising computer code that selectively transmits data from a software repository of the policy orchestrator server to the device via the agent corresponding to the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has with software transferred to the network device through the agent (column 11, line 40 – column 14, line 23, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

19. With regards to claim 32, Barroux teaches through Ries, the computer program product for management of a network of devices and resources via a computer network, further comprising computer code that transmits an agent installation module by the policy orchestrator server to the device for installation of the corresponding agent on the device

(Barroux teaches a system for managing a network of devices and resources. Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

20. With regards to claim 33, Barroux teaches through Ries, a computer program product for management of a network of devices and resources available to the devices via a computer network, comprising: computer code that contains a network directory defining a hierarchical tree structure containing nodes corresponding to the network of devices and defining policies corresponding to and to be enforced upon the resources available to the devices; computer code that facilitates communication between a policy orchestrator server and the network directory to facilitate accessing data from and storing data to the network directory, the data relating to the nodes of the hierarchical tree structure corresponding to the devices and to the policies corresponding to the resources; computer code that facilitates communication between an agent and the policy orchestrator server and the resources corresponding to the device, the agent computer code being adapted to enforce the policies corresponding to the resources contained in the network directory, wherein the policies corresponding to the resources of each device are selectively inherited down the hierarchical tree structure of the network directory; and a computer readable medium that stores said computer codes; and computer code that dynamically determines the policy of a selected node of the hierarchical tree structure by the management console; wherein the computer code that dynamically determines includes computer code that reads policies of nodes along a path of nodes from a path of nodes from a root of the hierarchical tree structure down to the selected node and computer code that overwrites previously written policies upon reading conflicting policies at each node along the path of nodes; wherein each

agent includes a task execution component for causing performance of a scheduled task at a scheduled time, a policy enforcement component for enforcing policies applicable to resources of the corresponding device, a property collection component for collecting and storing properties of the corresponding resources of the device and for transmitting the properties to the policy orchestrator server, and an event collection component for collecting and storing event data and for transmitting the event data to the policy orchestrator server

(Barroux teaches a system for managing a network of devices and resources. A GUI is provided to view the network from a hierarchical level (topology) (column 4, lines 49-52, Barroux). Barroux's design also allows for policy defining and enforcing (column 3, line 64 – column 4, line 14, Barroux). This includes the search of the network and policy handling techniques claimed. In addition, Barroux's design has agents within each device to assist in monitoring and managing the network (column 3, lines 41-53, Barroux). Plus in Barroux's design the processes to be performed, are computed (column 4, line 3, Barroux), hence they are determined dynamically. In addition, it is inherent that nodes will be check along the hierarchical path. Barroux's design checks nodes individually along the network path (column 14, lines 25 – column 18, line 21, Barroux). Finally, Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. In addition, Barroux's design allows for scheduled tasks (column 3, line 64 – column 4, line 13, Barroux) and device property retrieval and storage as claimed (column 14, line 25 – column 18, line 21, Barroux). However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

23. With regards to claim 34, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, wherein a directory management display of the management console includes a scope pane, a selected node directory pane, and a details pane

(Barroux teaches a system for managing a network of devices and resources. Barroux's design uses a GUI (Figures 6A, 6B and 6C, Barroux). In addition, Barroux discloses that further embodiments in view of the descriptions of a disclosure and apparent to those skilled in the art are permissible (column 19, lines 54-60, Barroux). Adjustments to the GUI are reasonable apparent changes to one skilled in the art and hence it is acceptable. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries). Furthermore, Ries discloses a design with a GUI with multiple panes (Figure 8, Ries). The left most pane allows users to view the network from a wide scope view. After a selection is made in the left most pane, it's immediate right pane displays elements selectable by the user for viewing. Further right is a pane that displays groups within the selected elements that are viewable and the right most pane displays details regarding the final element selected. The use of panes with different scopes to view a network's elements with, as claimed, is common in the art and is also present in Ries' design.

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

24. With regards to claim 35, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, wherein the scope pane displays the hierarchical tree structure as populated by the management console

(Barroux teaches a system for managing a network of devices and resources. Barroux's design uses a GUI (Figures 6A, 6B and 6C, Barroux). In addition, Barroux discloses that further embodiments in view of the descriptions of a disclosure and apparent to those skilled in the art are permissible (column 19, lines 54-60, Barroux). Adjustments to the GUI are reasonable apparent changes to one skilled in the art and hence it is acceptable. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries). Furthermore, Ries discloses a design with a GUI with multiple panes (Figure 8, Ries). The left most pane allows users to view the network from a wide scope view. After a selection is made in the left most pane, it's immediate right pane displays elements selectable by the user for viewing. Further right is a pane that displays groups within the selected elements that are viewable and the right most pane displays details regarding the final element selected. The use of panes with different scopes to view a network's elements with, as claimed, is common in the art and is also present in Ries' design. Furthermore, the use of hierarchical tree structures is also common, as is in Ries' design (Figure 3, Ries).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

25. With regards to claim 36, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, wherein the scope pane includes a software tab and a directory tab for display of software and director content in the scope pane

(Barroux teaches a system for managing a network of devices and resources. Barroux's design uses a GUI (Figures 6A, 6B and 6C, Barroux). In addition, Barroux discloses that further embodiments in view of the descriptions of a disclosure and apparent to those skilled in the art are permissible (column 19, lines 54-60, Barroux). Adjustments to the GUI are reasonable apparent changes to one skilled in the art and hence it is acceptable. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries). Furthermore, Ries discloses a design with a GUI with multiple panes (Figure 8, Ries). The left most pane allows users to view the network from a wide scope view. After a selection is made in the left most pane, it's immediate right pane displays elements selectable by the user for viewing. Further right is a pane that displays groups within the selected elements that are viewable and the right most pane displays details regarding the final element selected. The use of panes with different scopes to view a network's elements with, as claimed, is common in the art and is also present in Ries' design. Furthermore, the uses of tabs or similar interfaces (such as buttons and menus (Figure 3, Ries)) are also common in software applications with GUIs.

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

26. With regards to claim 37, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, wherein the selected node directory pane includes a policies tab, a properties tab, an events tab, and a tasks tab for display of policies, properties, events, and tasks in the selected node directory pane

(Barroux teaches a system for managing a network of devices and resources. Barroux's design uses a GUI (Figures 6A, 6B and 6C, Barroux). In addition, Barroux discloses that further embodiments in view of the descriptions of a disclosure and apparent to those skilled in the art are permissible (column 19, lines 54-60, Barroux). Adjustments to the GUI are reasonable apparent changes to one skilled in the art and hence it is acceptable. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries). Furthermore, Ries discloses a design with a GUI with multiple panes (Figure 8, Ries). The left most pane allows users to view the network from a wide scope view. After a selection is made in the left most pane, it's immediate right pane displays elements selectable by the user for viewing. Further right is a pane that displays groups within the selected elements that are viewable and the right most pane displays details regarding the final element selected. The use of panes with different scopes to view a network's elements with, as claimed, is common in the art and is also present in Ries' design. Furthermore, the uses of tabs or similar interfaces (such as buttons and menus (Figure 3, Ries)) are also common in software applications with GUIs.

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

27. With regards to claim 38, Barroux teaches through Ries, the method for management of a network of devices and resources via a computer network, wherein, if a node is selected, the node is highlighted in the hierarchical tree structure in the scope pane and details of the hierarchical tree structure and software hierarchy for the selected node is displayed in the selected node directory pane

(Barroux teaches a system for managing a network of devices and resources. Barroux's design uses a GUI (Figures 6A, 6B and 6C, Barroux). In addition, Barroux discloses that further embodiments in view of the descriptions of a disclosure and apparent to those skilled in the art are permissible (column 19, lines 54-60, Barroux). Adjustments to the GUI are reasonable apparent changes to one skilled in the art and hence it is acceptable. However, Barroux does not teach the use of inherited policies.

Ries also teaches a design for monitoring a network. Ries' design though does allow for policies to be inherited (column 3, lines 13-16, Ries). Furthermore, Ries discloses a design with a GUI with multiple panes (Figure 8, Ries). The left most pane allows users to view the network from a wide scope view. After a selection is made in the left most pane, it's immediate right pane displays elements selectable by the user for viewing. Further right is a pane that displays groups within the selected elements that are viewable and the right most pane displays details regarding the final element selected. The use of panes with different scopes to view a network's elements with, as claimed, is common in the art and is also present in Ries' design. Furthermore, just as is common in most software applications with GUIs, selection of items are highlighted (Figures 3 and 8, Reis).

While both Barroux's design and Ries' design disclose methods for network monitoring, it is Ries' design that teaches the use of inherited policies. It would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Ries with those of Barroux, for the purpose of measuring performance and monitoring service quality within an information system (column 2, lines 62-65, Ries)).

Response to Remarks

The arguments filed by the applicant's representative on May 28, 2004 have been thoroughly considered but they are not deemed fully persuasive. The following are brief explanations in response to the arguments presented.

The applicant's representative remarks that the agents in the prior arts presented vary from the agents of the claimed design. The difference between the agents being that the prior art agents does not enforce policies corresponding to the resources. The term "enforce" is broad and does not accurately portray what steps are actually performed by the agents. The agents of Barroux's design (column 18, lines 40-53, Barroux) perform tasks such as retrieve and send messages corresponding to each of the devices in the network system to assist in performing the network system's tasks. These steps are a form of enforcement corresponding to resources as claimed.

The remark concerning overwriting previously written policies when conflicting policies are detected upon reading policies of each node along the path was also considered by the examiner. After review of the art though, this as well was disagreed with. The Barroux art uses tree architecture with nodes (devices). Hence when a network is searched, each of the nodes of a path along a tree where the desired device is, is read as claimed. It is common in computer designs when something read is conflicting with what is to be written, that the previous data read will be overwritten. This is common practice in systems using memory to prevent wastage of memory. This is a practice in the Barroux design as well. Barroux's design allows administrative changes to be overwritten (column 7, lines 32-41, Barroux). It is for such reasoning that the claimed traits are considered inherent. The traits are practiced in many levels in the computer field.

Finally, the remarks concerning an event collection component for collecting and storing event data for transmitting the event data to the policy orchestrator server was also disagreed to by the examiner after careful consideration. As stated, Barroux's design has a software repository and allows for the software to be delivered and installed into the network devices (column 11, line 40 – column 14, line 22, Barroux). An agent must be present for such a process to occur as claimed. It must be present because it must inform the network management system of any errors encountered or the successful completion of the task. In addition, Barroux's design allows for scheduled tasks (equivalent to events) (column 3, line 64 – column 4, line 13, Barroux) and device property retrieval and storage as claimed (column 14, line 25 – column 18, line 21, Barroux).

The claimed design as a whole, as currently stated describes a network management/monitoring design. The traits within the claims are commonly available in network management/monitoring designs such as the prior arts presented. It should be kept in mind that the prior arts should not be taken simply literally but must also take into account, the spirit of the design being described. With such an understanding of the prior art, the examiner believes the current claims fail to describe a design that distinguishes itself from the designs of the prior art.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Azizul Choudhury whose telephone number is 703-305-7209. The examiner can normally be reached on M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David Wiley can be reached on 703-308-5221. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AC

DAVID WILEY
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100